Pivotal

Day 2 Kubernetes
Beyond the install

Paul Czarkowski
Principal Technologist

@pczarkowski
github.com/paulczar

ELECTRONIC ARTS™

Pivotal™

Pivotal


I'M AN
DEVLOPER AVOCADO
memegen.com


Who is using Linux Containers ?
Pivotal


Who is using Kubernetes ?


Who is operating Kubernetes ?


> 1 year Kubernetes experience?

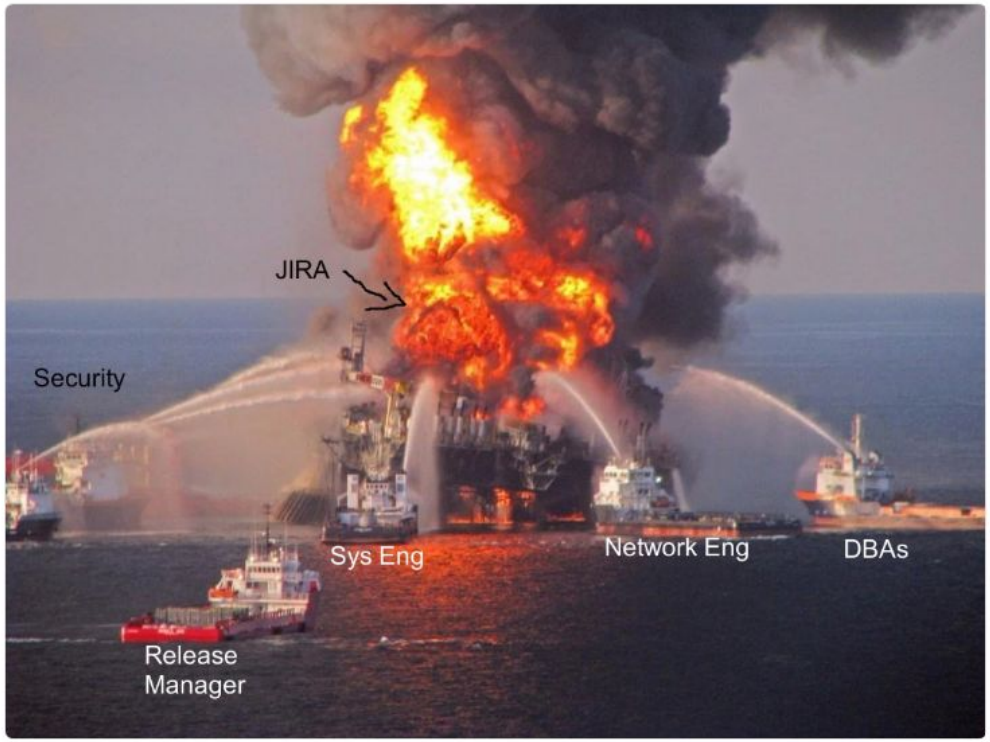
Successful Kubernetes deployments are just as much a cultural challenge as they are technical.

heptio

KubeCon | CloudNativeCon Europe 2018

Czarcloudski
@pczarkowski

Enterprise DevOps



11:16 AM - 11 Jun 2018

662 Retweets **1,315** Likes

Pivotal

People build Platforms
People build Apps
Apps run on Platforms

People are the most important component of any platform.


“Organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations.” - Conways Law


OBSERVE
ORIENT
the
OODA
LOOP
DECIDE
ACT

Control loops

Drive **current state** -> **desired state**

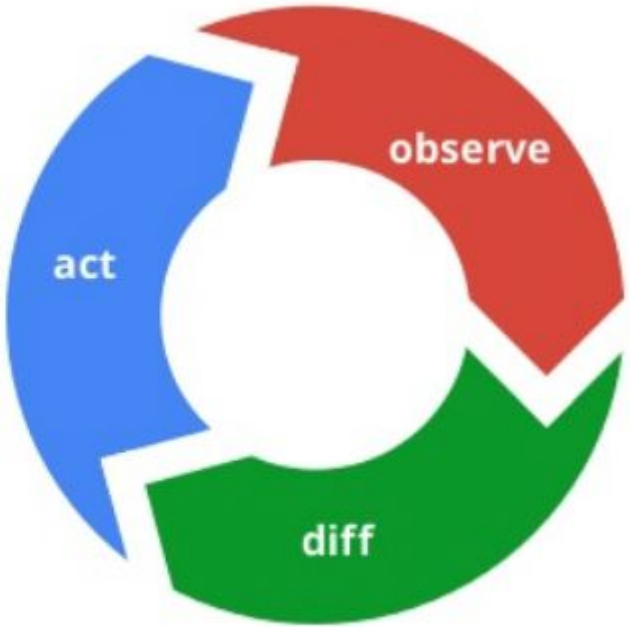
Act independently

APIs - **no shortcuts** or back doors

Observed state is truth

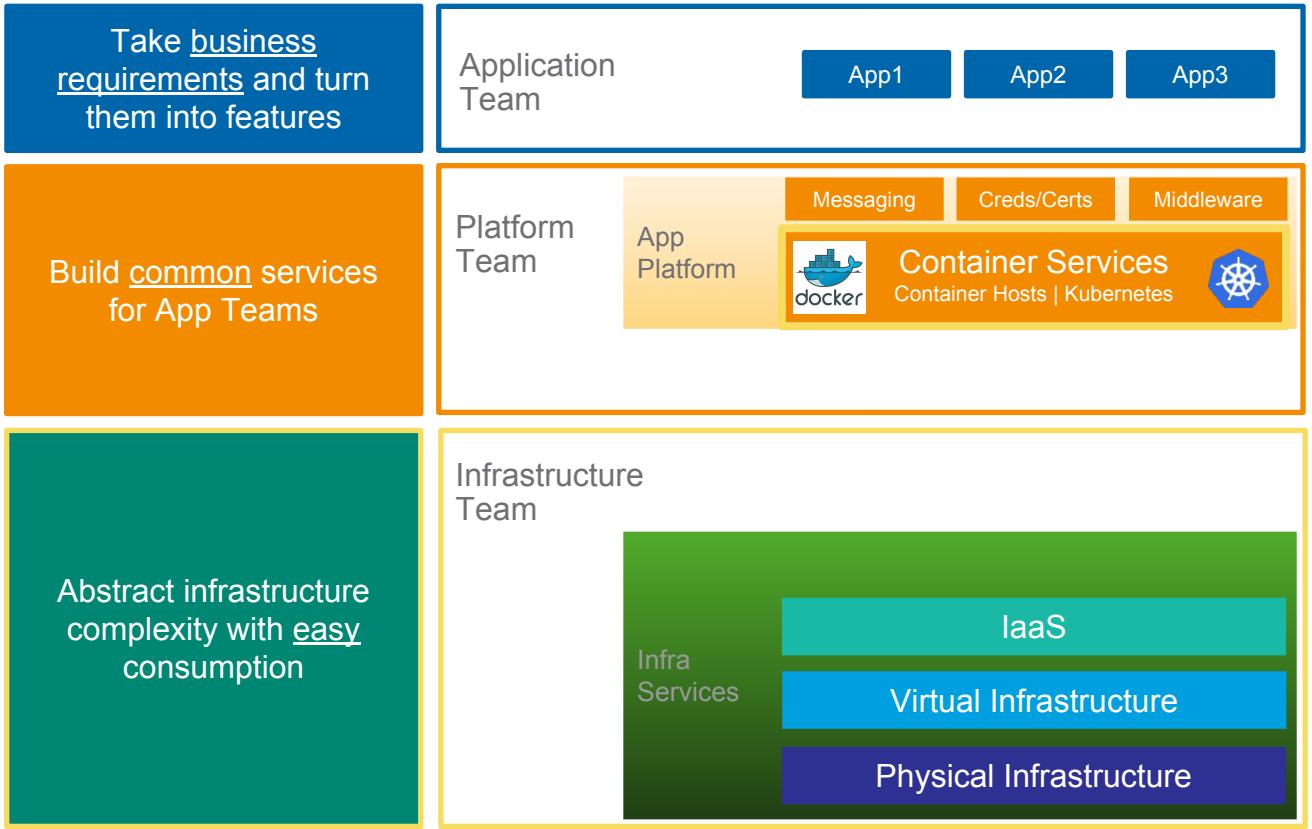
Recurring pattern in the system

Example: ReplicationController

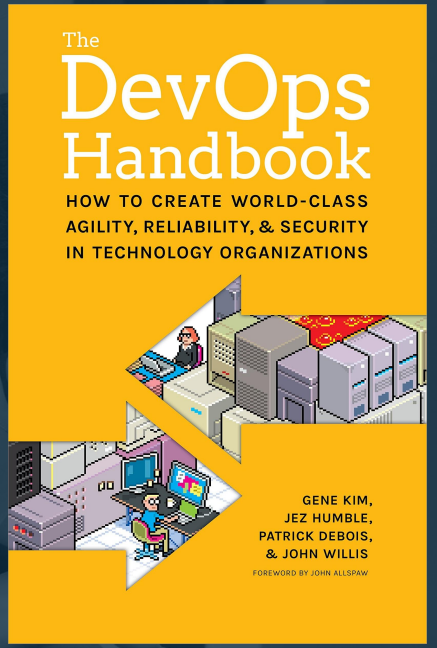


https://www.slideshare.net/SatnamSingh67/2015-0605-cluster-management-with-kubernetes

Evolve your IT teams!



"In general, taking something that's already working somewhere and expanding its usage (capabilities) is far more likely to succeed than building these capabilities from scratch"



Day 1 - Build

Development The team can make progress in developing new features for the platform

CI/CD CI/CD pipelines drive the testing and promotion of artifacts

Consistency Provide a consistent setup experience, across different environment configurations.

Setup time How long does it take to setup a real world working environment? Think hours, not weeks.

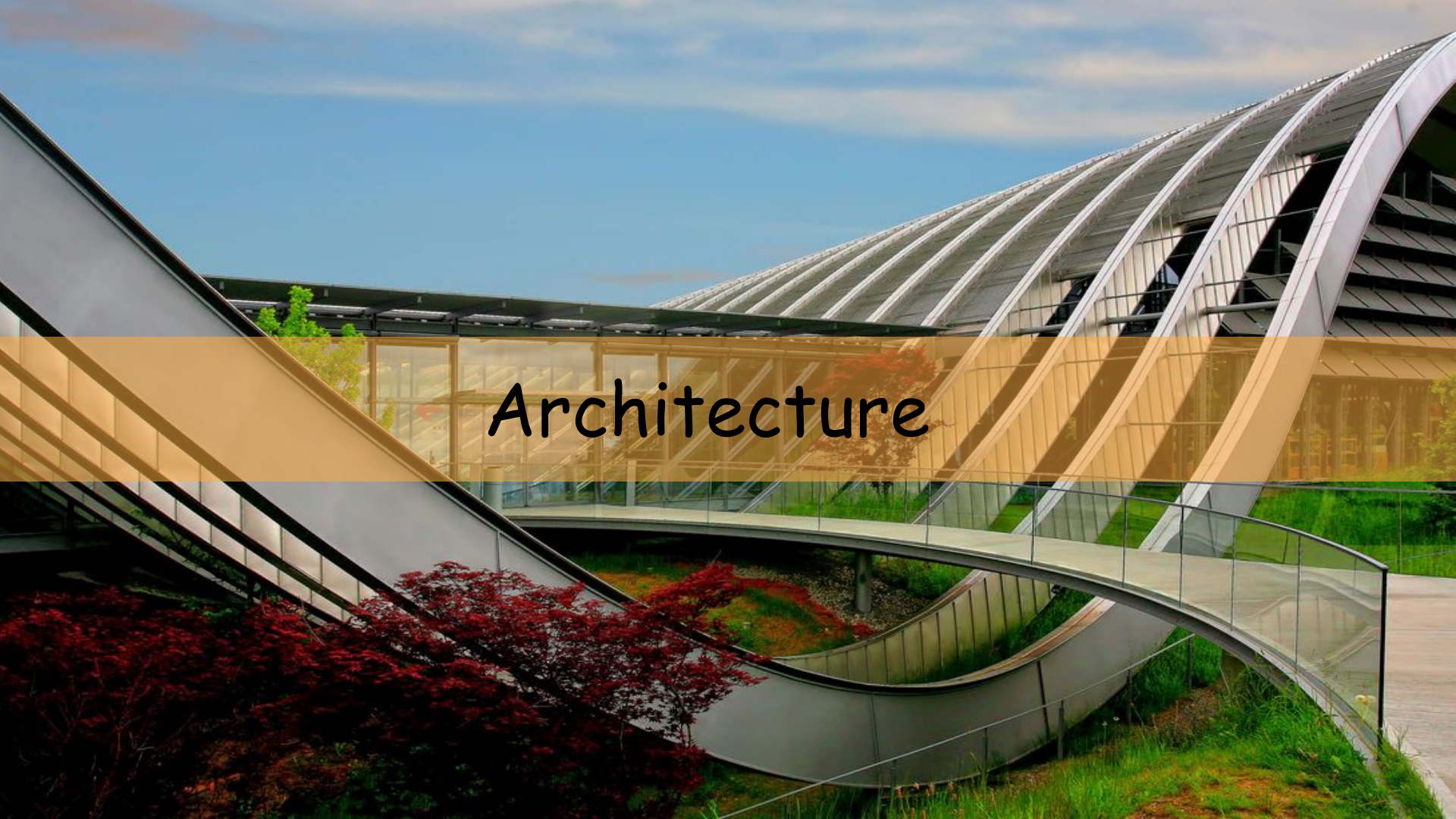
Day 2 - Operate & Enhance

Patches Patching App and System components as CVEs occur

Scaling Seamlessly scale platform components to accommodate changing demand.

Upgrades. How do you roll out new versions of the platform with the lights on?

Operating Effort Operating the platform should require very few resources and minimum manual intervention. Otherwise, you will be spending lots on operational support!

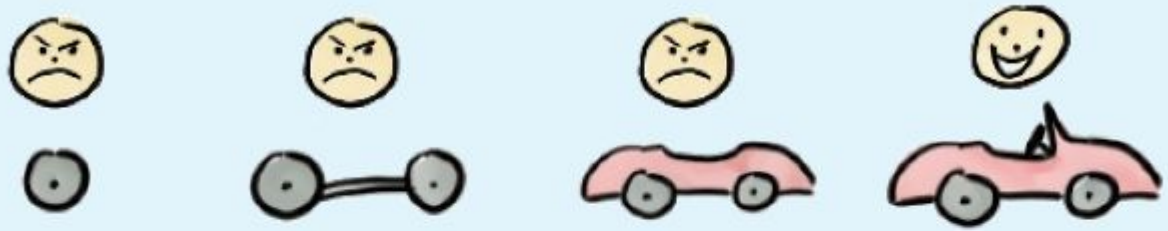

Architecture

Czarcloudski
@pczarkowski

Cant tell if london underground map or openstack architecture diagram.

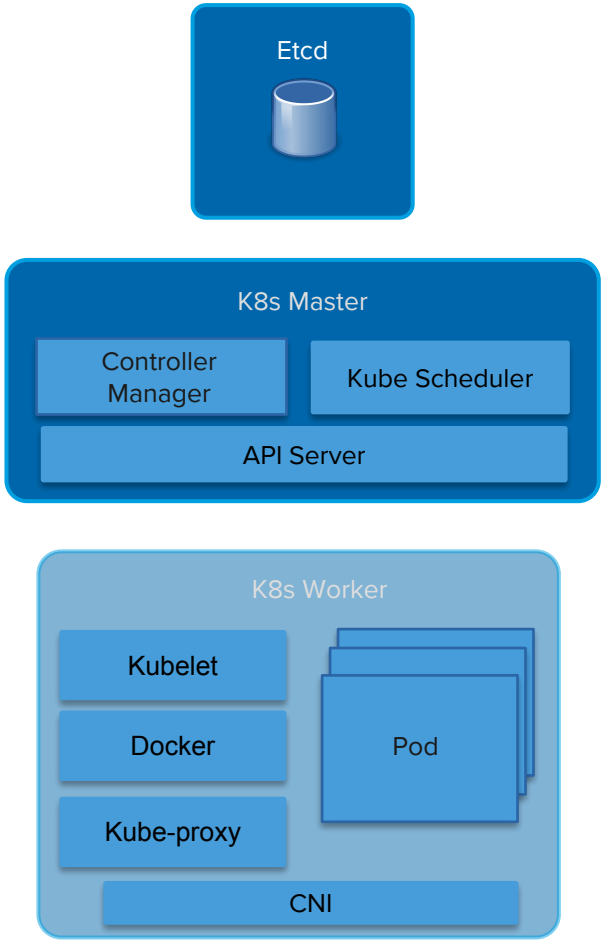
7:43 AM - 19 Feb 2016

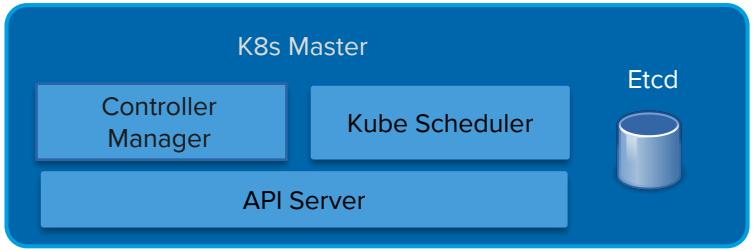
Not like this...

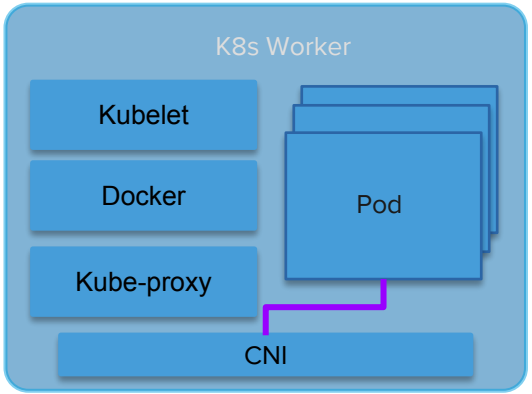


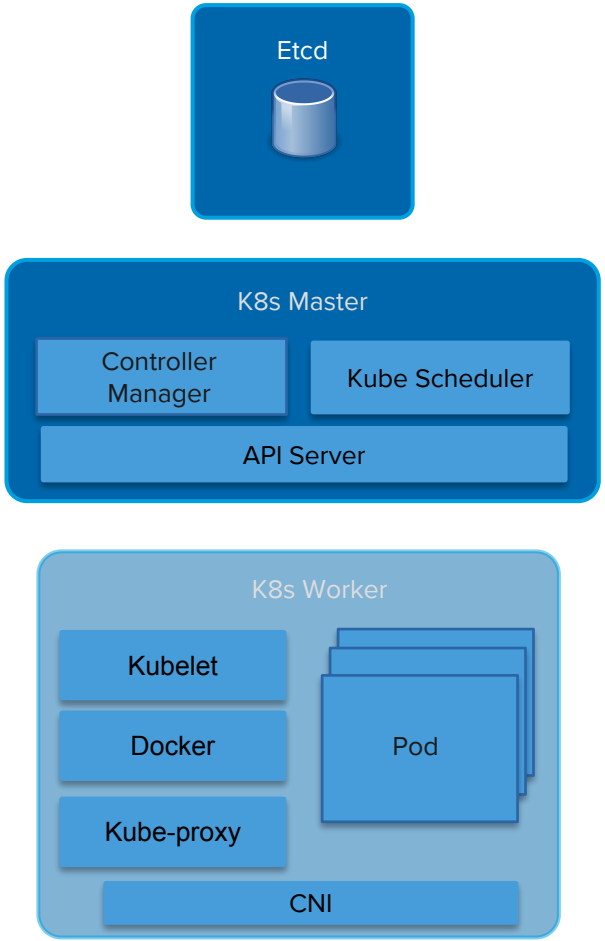
Like this...

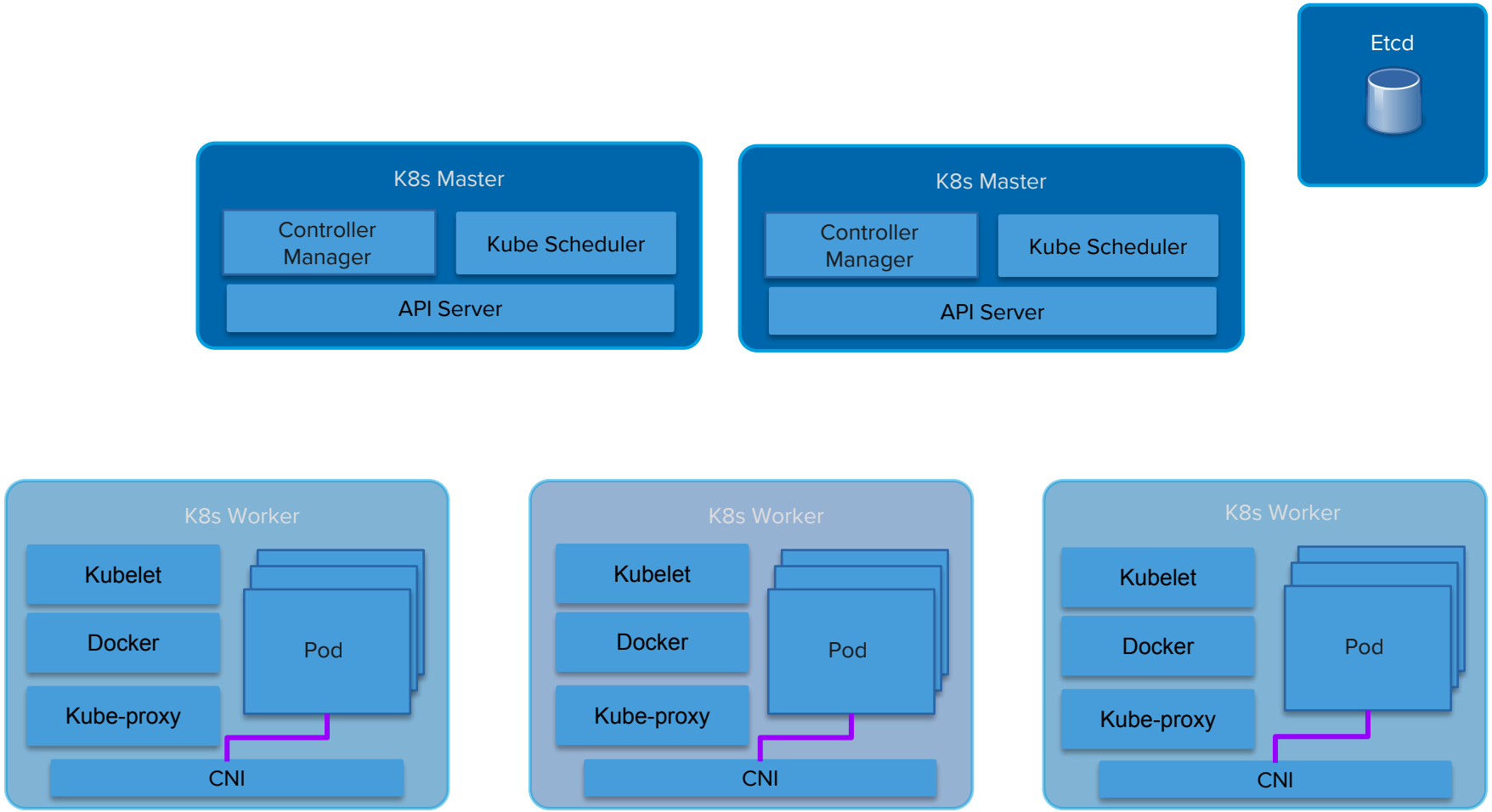
Created by Henrick Kniberg

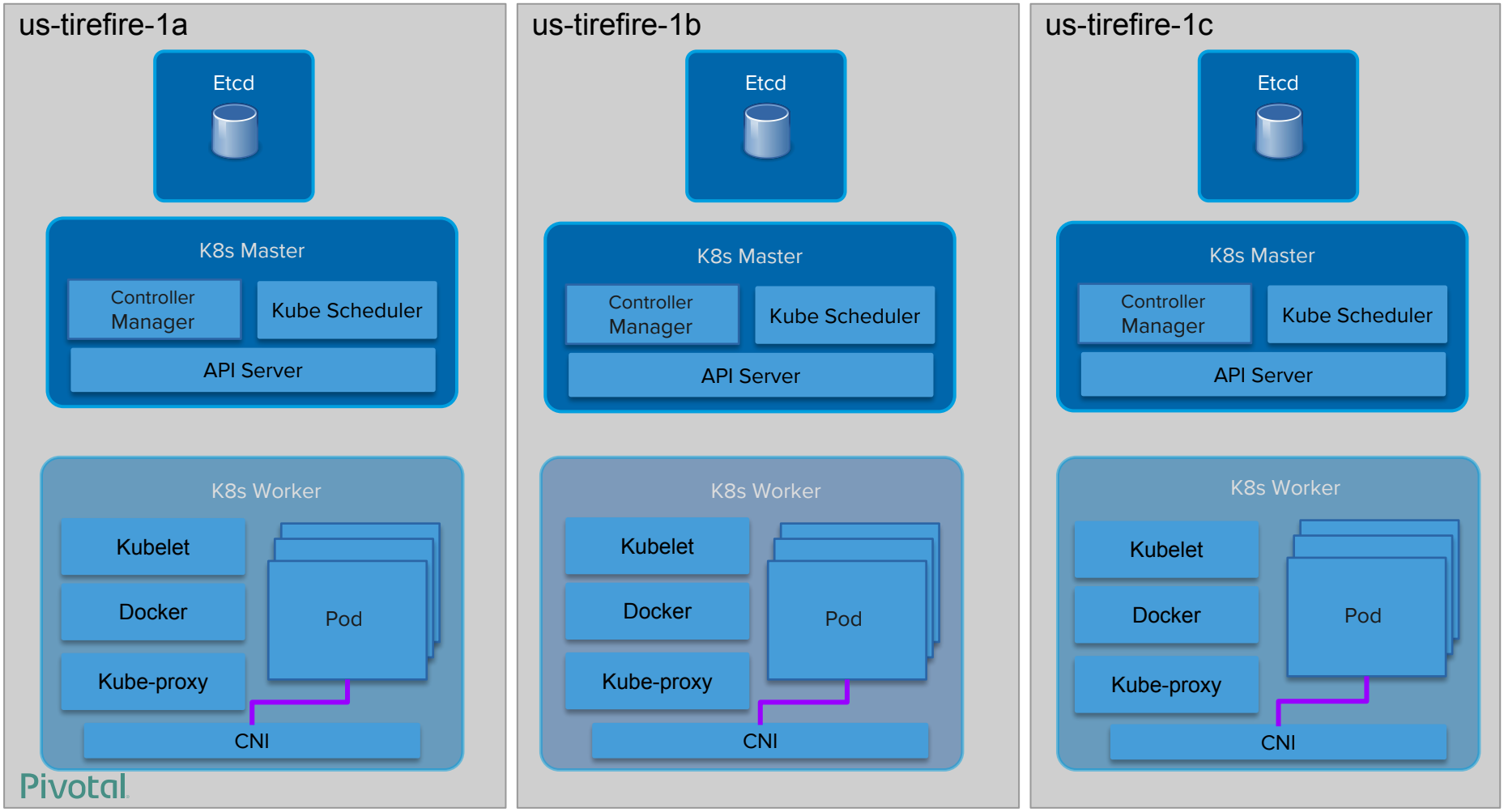

Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI


K8s Master
Controller Manager
Kube Scheduler
API Server
Etcd

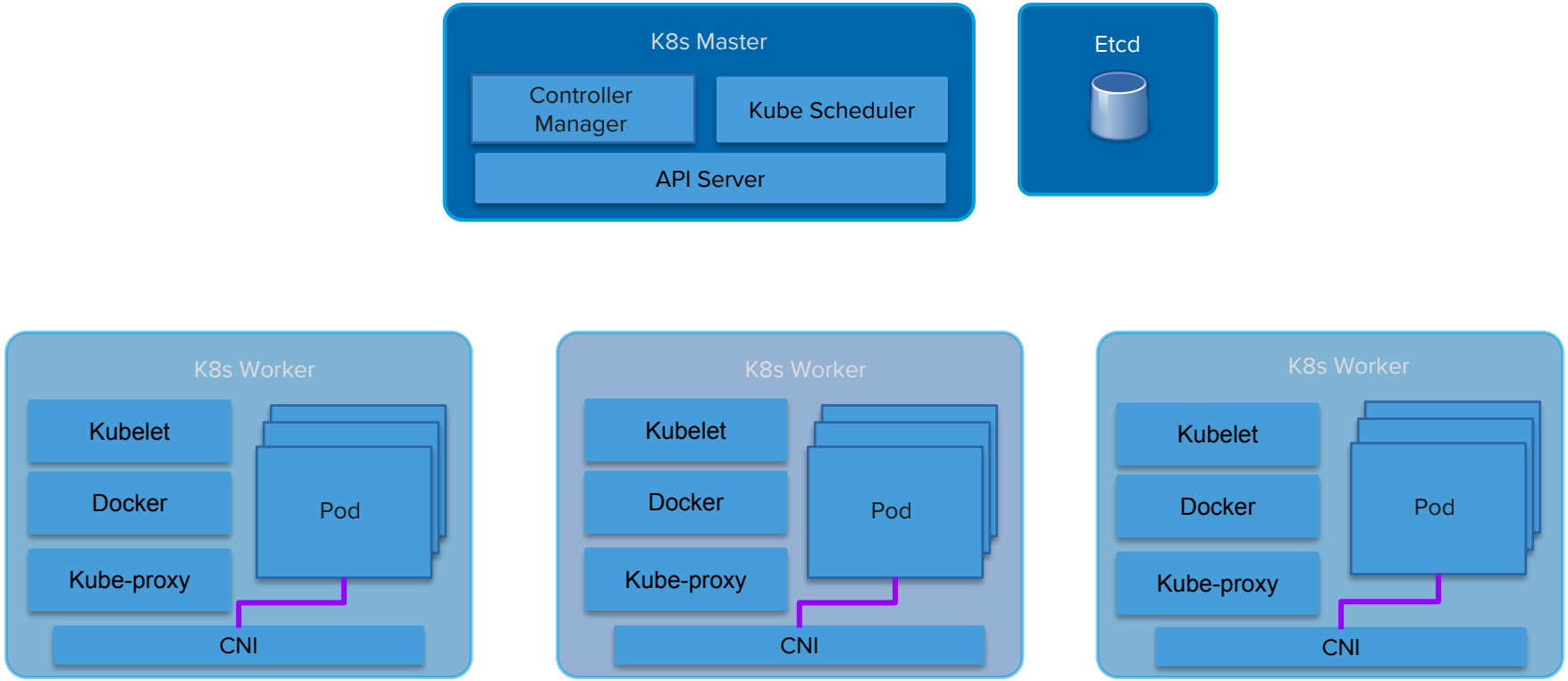

K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI

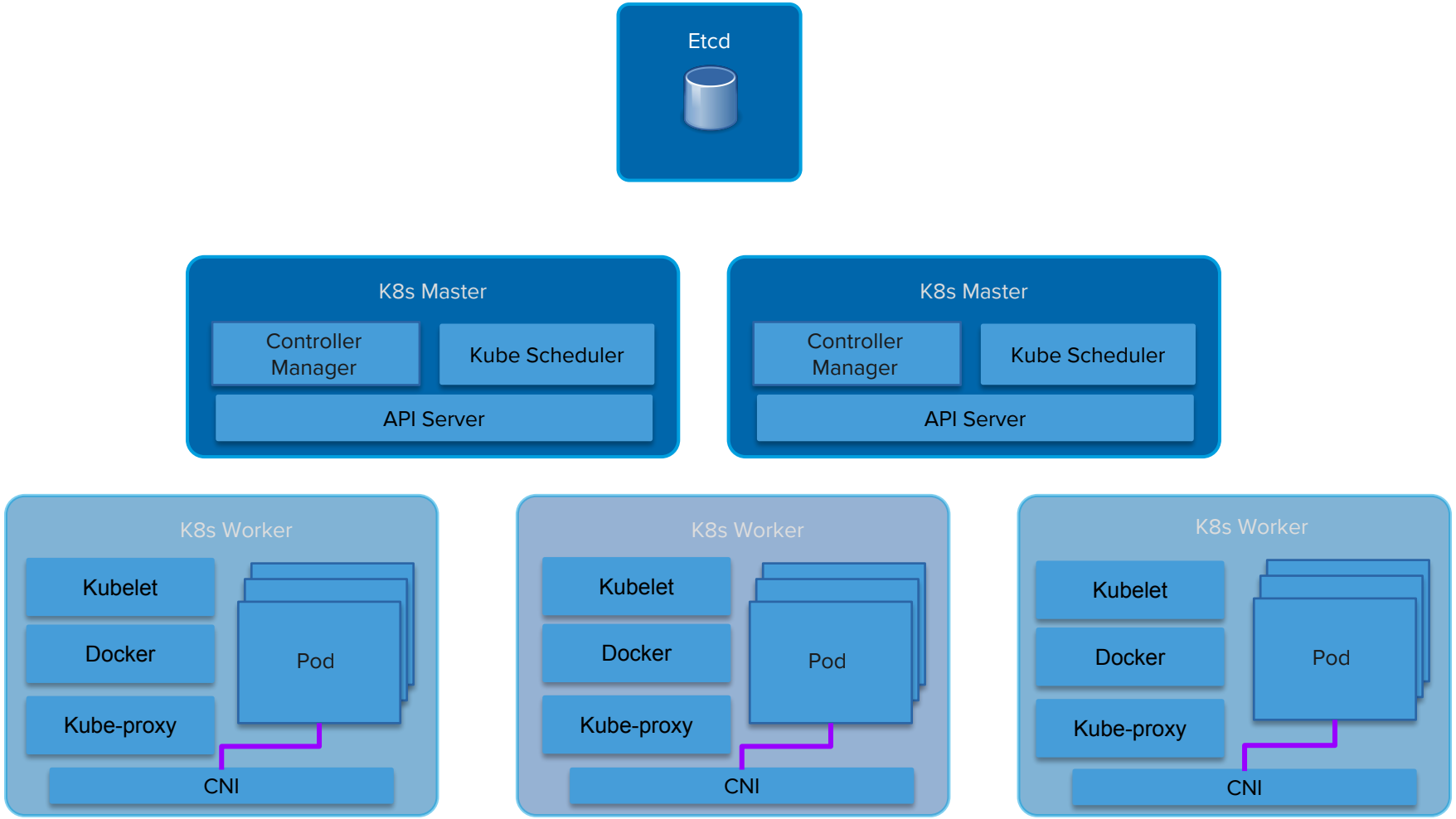

Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI

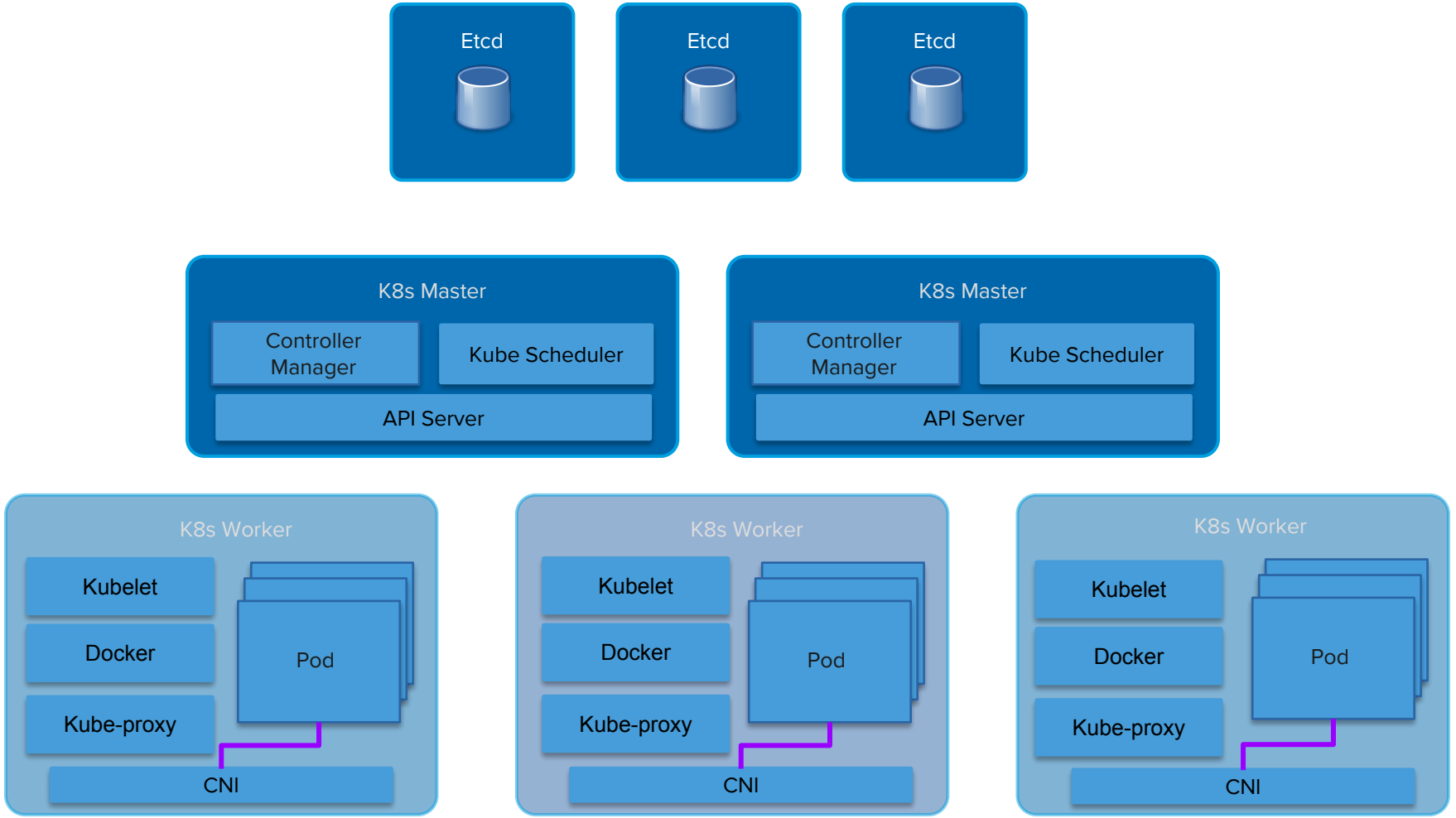

Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI

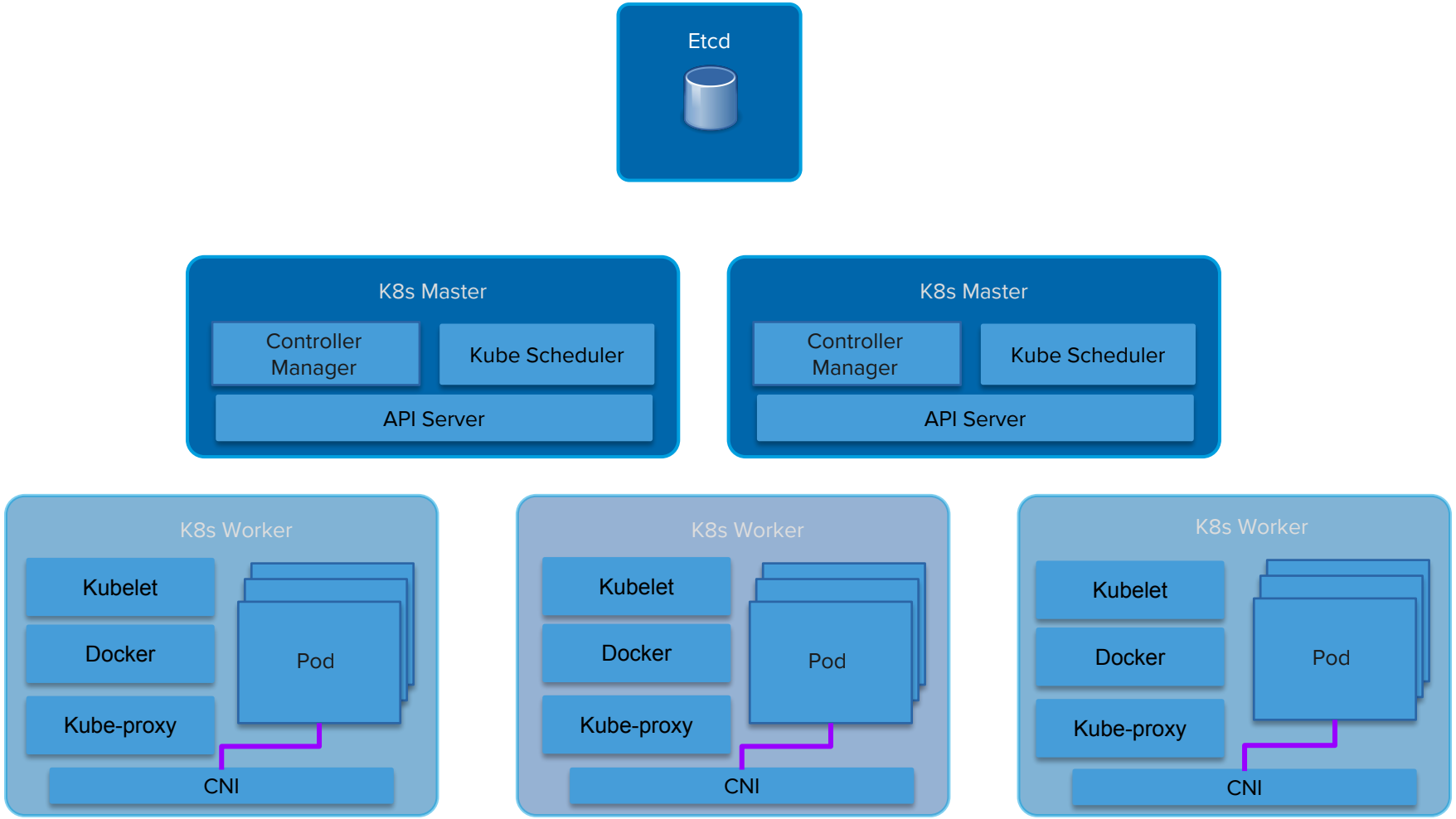

us-tirefire-1a
Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1b
Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal

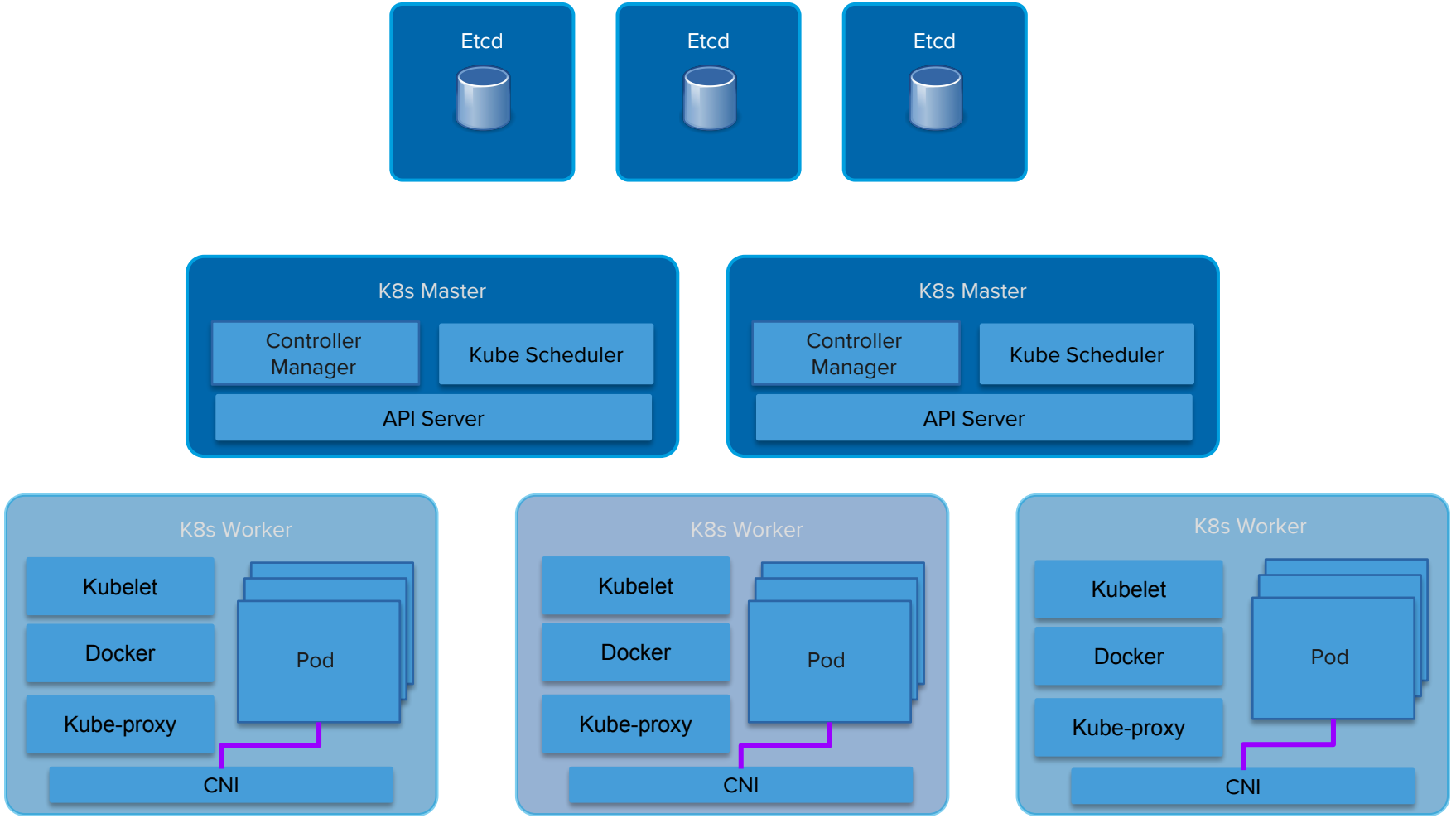

High Availability


K8s Master
Controller Manager
Kube Scheduler
API Server
Etcd
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI

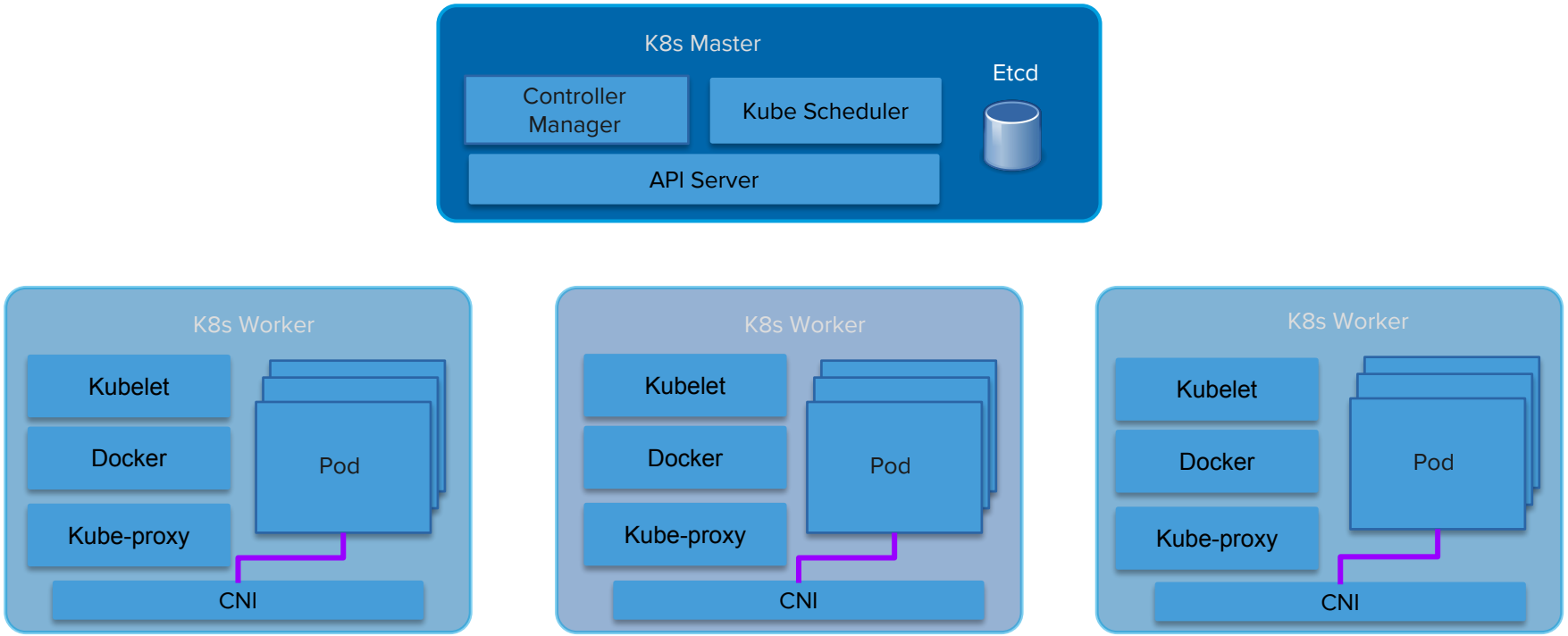

Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI


Etcd
Etcd
Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI


Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI


Etcd
Etcd
Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI

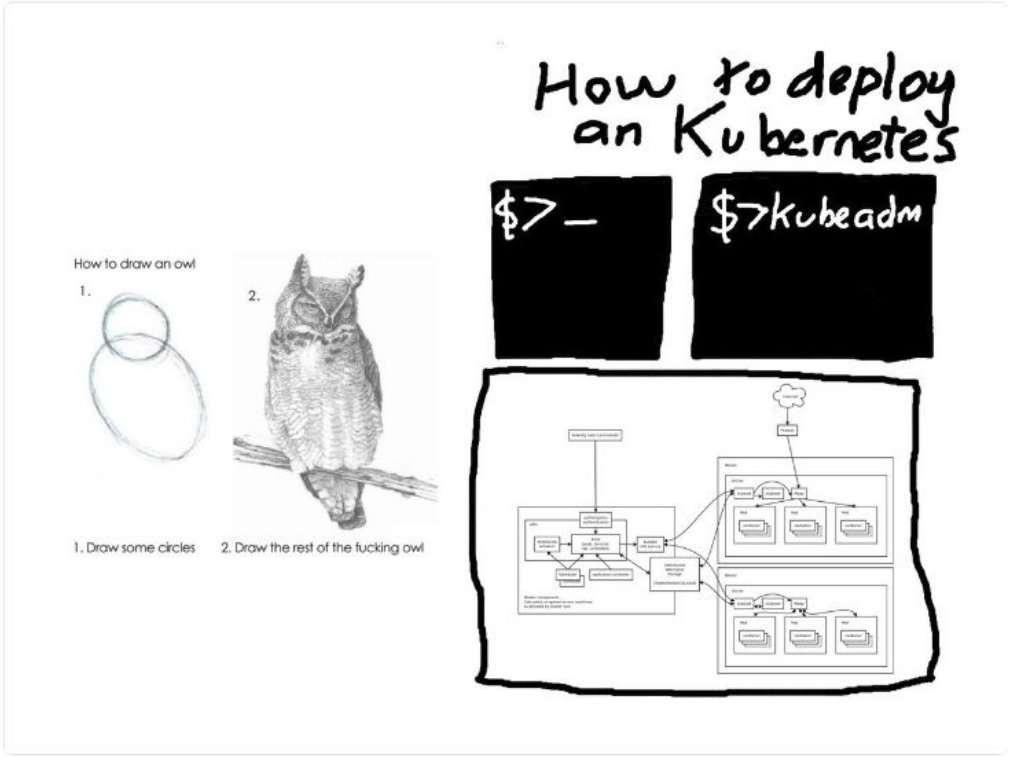
http://www.bsielearning.com.au/keep-simple-stupid/


K8s Master
Controller Manager
Kube Scheduler
API Server
Etcd
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI

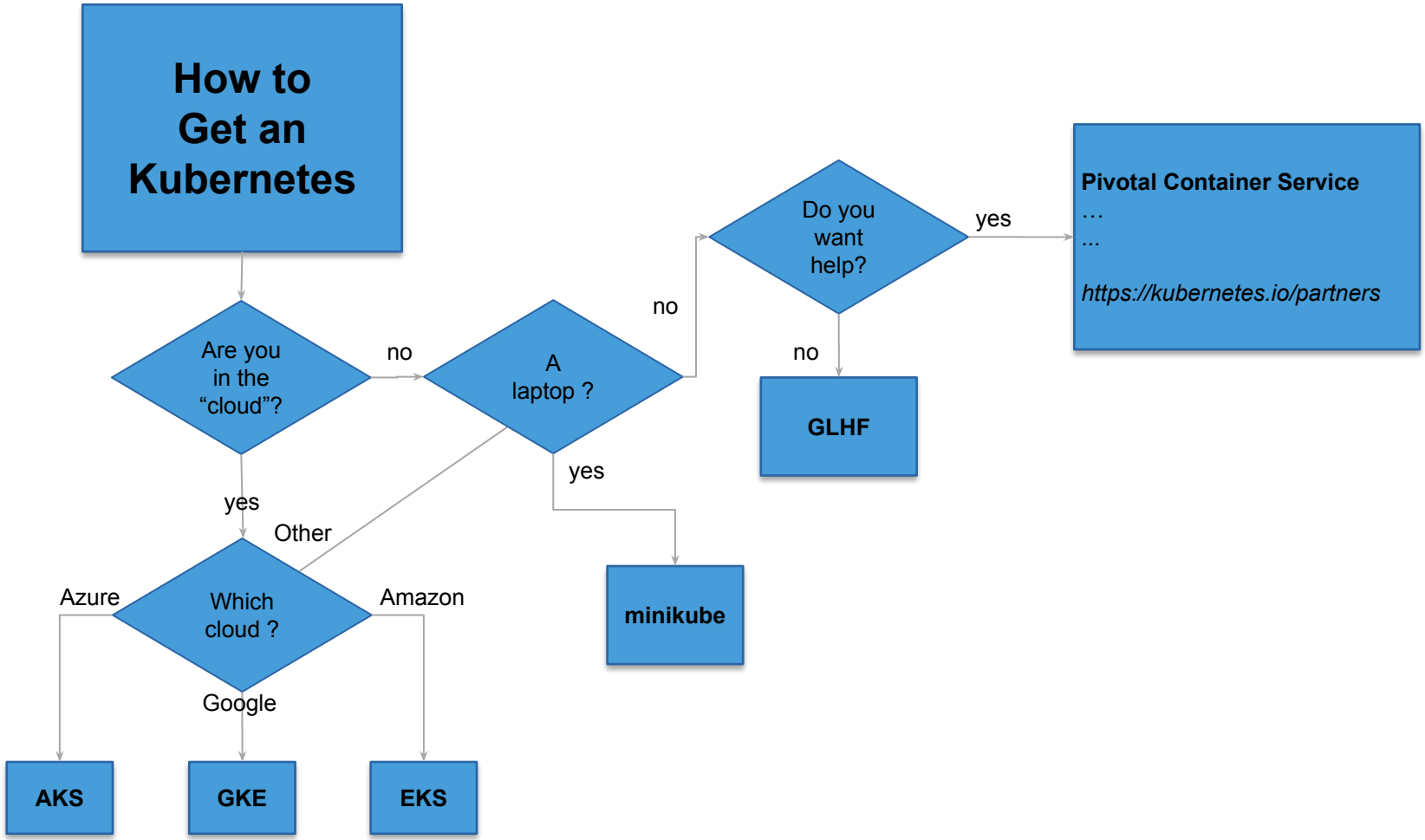

Deployment

Czarcloudski
@pczarkowski

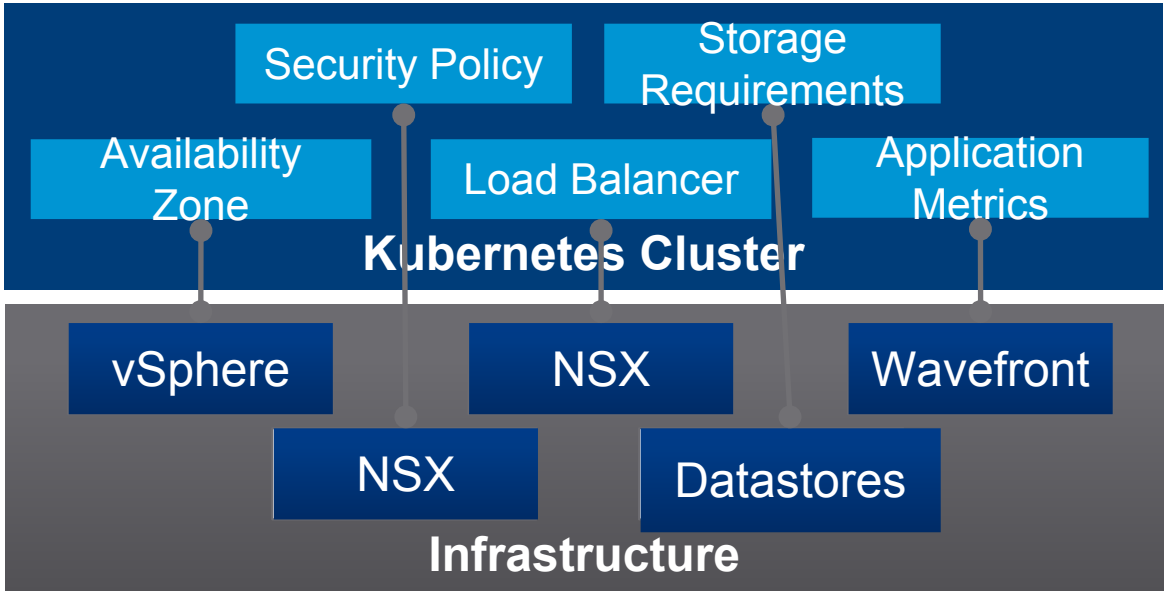
Did you know that Deploying an Kubernetes is as simple as drawing an Owl ?

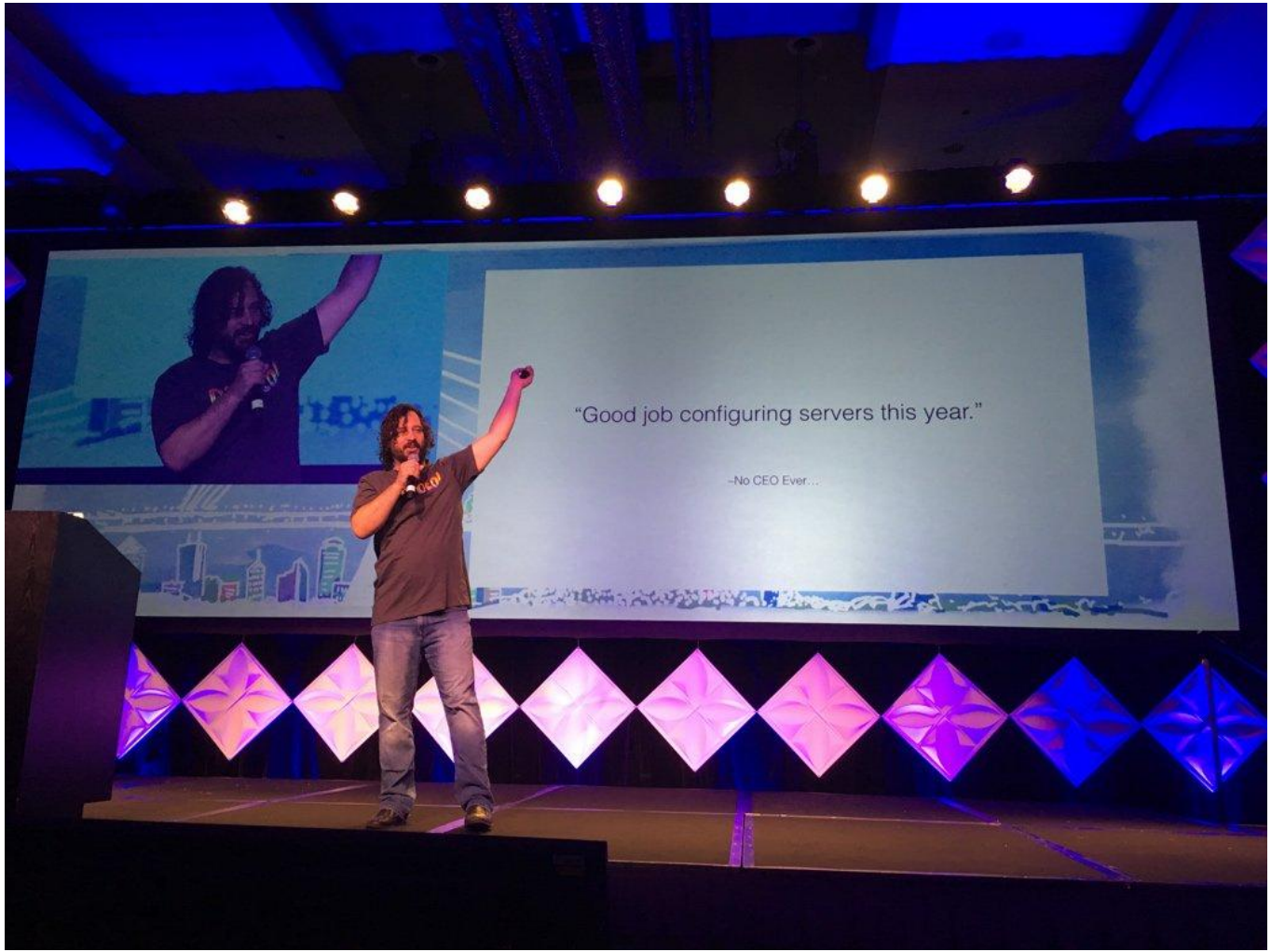


6:05 PM - 18 Jul 2017


How to Get an Kubernetes
Are you in the "cloud"?
no
A laptop ?
Do you want help?
yes
Pivotal Container Service
…
...
https://kubernetes.io/partners
no
GLHF
yes
minikube
yes
Other
Which cloud ?
Azure
Amazon
Google
AKS
GKE
EKS


vmware®


Security Policy
Storage Requirements
Availability Zone
Load Balancer
Application Metrics
Kubernetes Cluster
vSphere
NSX
Wavefront
NSX
Datastores
Infrastructure


"Good job configuring servers this year."
-No CEO Ever…

https://docs-cfcr.cfapps.io/

https://github.com/openshift/origin

https://github.com/kubernetes-incubator/kubespray

Kubespray

https://github.com/kubernetes-incubator/kubespray

- Ansible based, so very approachable
- An official Kubernetes (incubator) project
- Good support for CNIs and Cloud Providers
- Combine with one of the Ansible Hardening projects
  - https://github.com/dev-sec/ansible-os-hardening
  - https://github.com/openstack/ansible-hardening

gitops

- Deployed Platform == code repo + environment repo
  - Ansible - Playbook + Inventory
  - Bosh - Release + Manifest
- Keep it all in git!
  - Fork upstream repo... if only to ensure it doesn't get changed from under you
  - Inventory/Manifest is probably YAML ... perfect to be stored in git.
  - One repo for all envs, or a repo per env ... either is fine.
- Consider using a gitops focussed wrapper around ansible
  - Ursula-cli (https://github.com/blueboxgroup/ursula-cli)
  - Gosible (https://github.com/paulczar/gosible)
  - Molecule (https://github.com/metacloud/molecule)
- Use Jenkins or similar to run tests, deploy test envs, push to prod???
  - But probably not full on Continuous Delivery ... risks are very high!

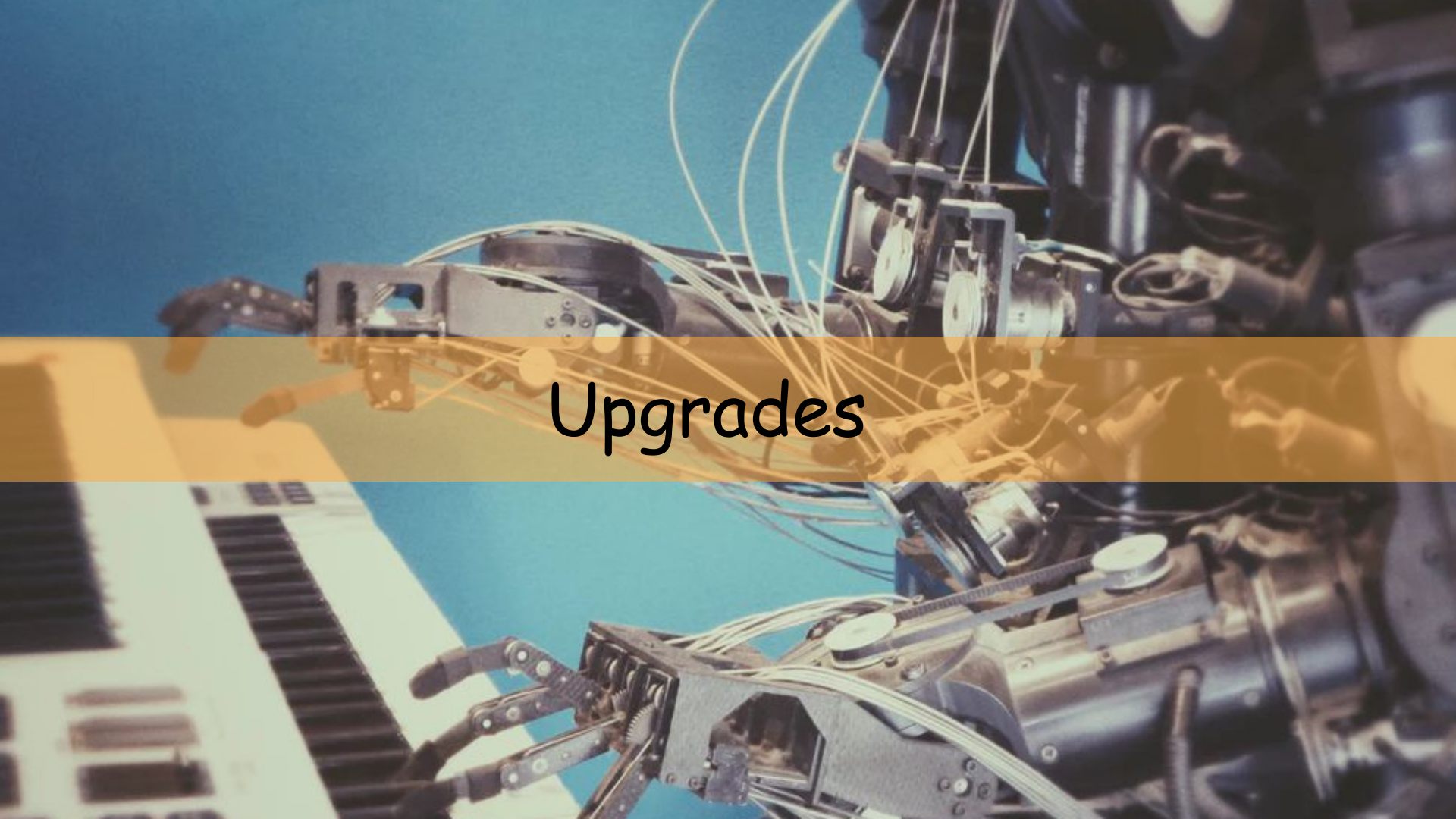
Validate and Backup

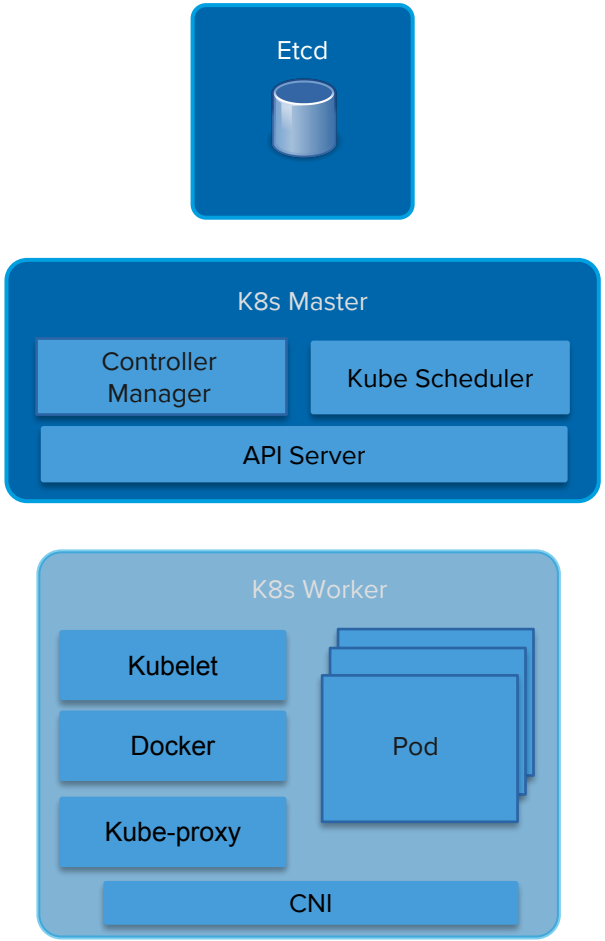
Validate your Kubernetes cluster is conformant!

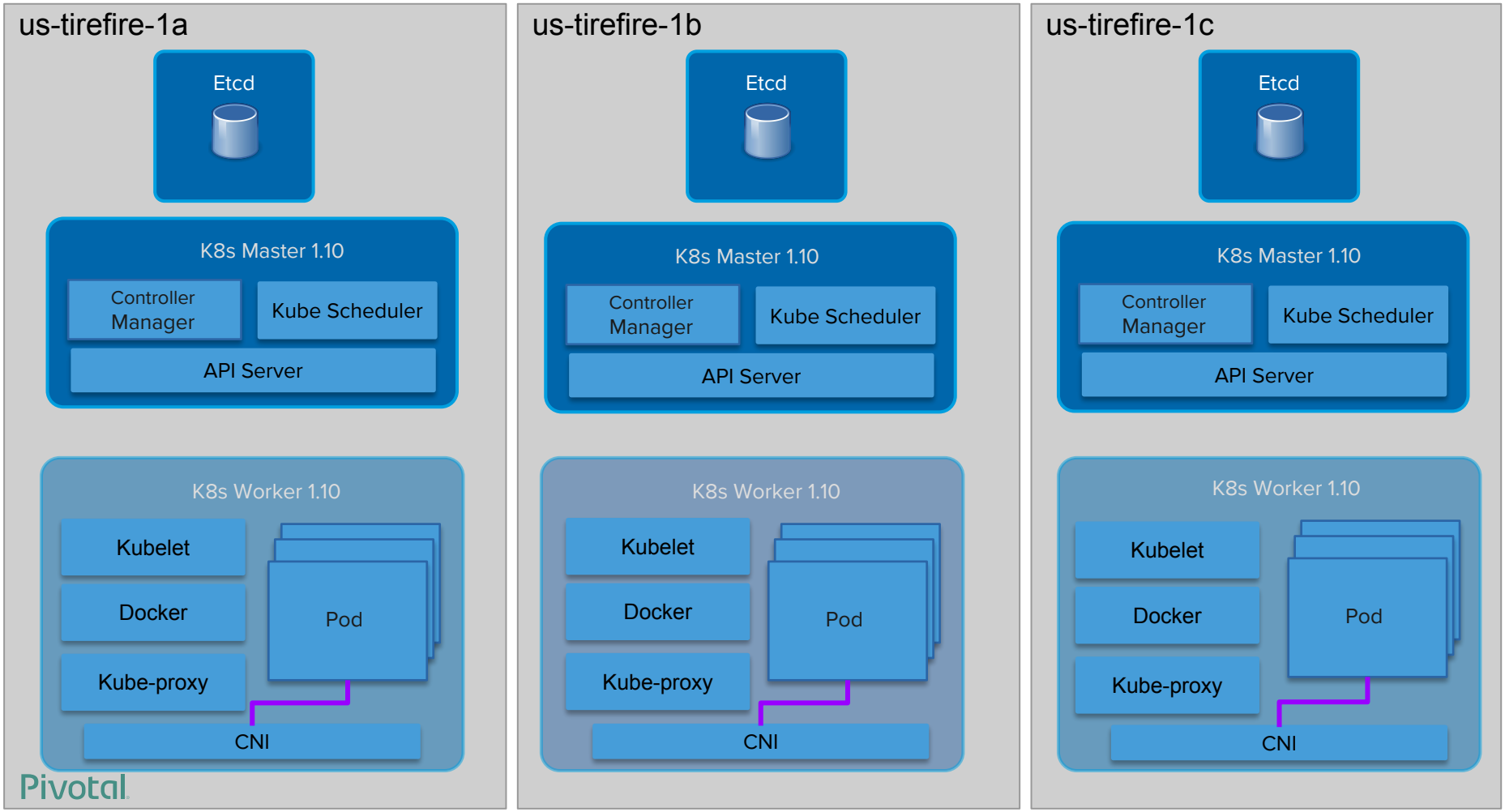
https://github.com/heptio/sonobuoy

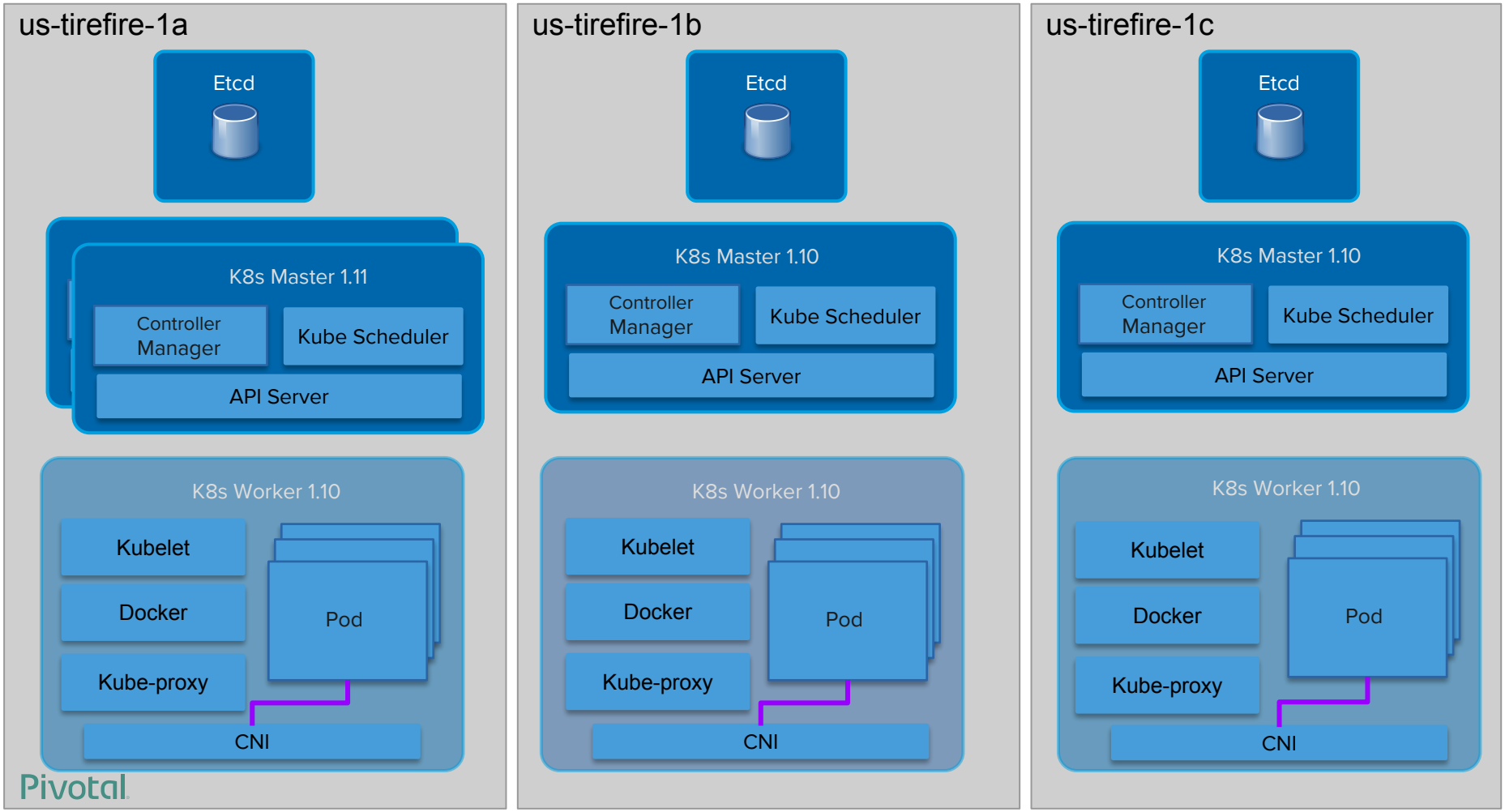
Backup your Kubernetes cluster state!

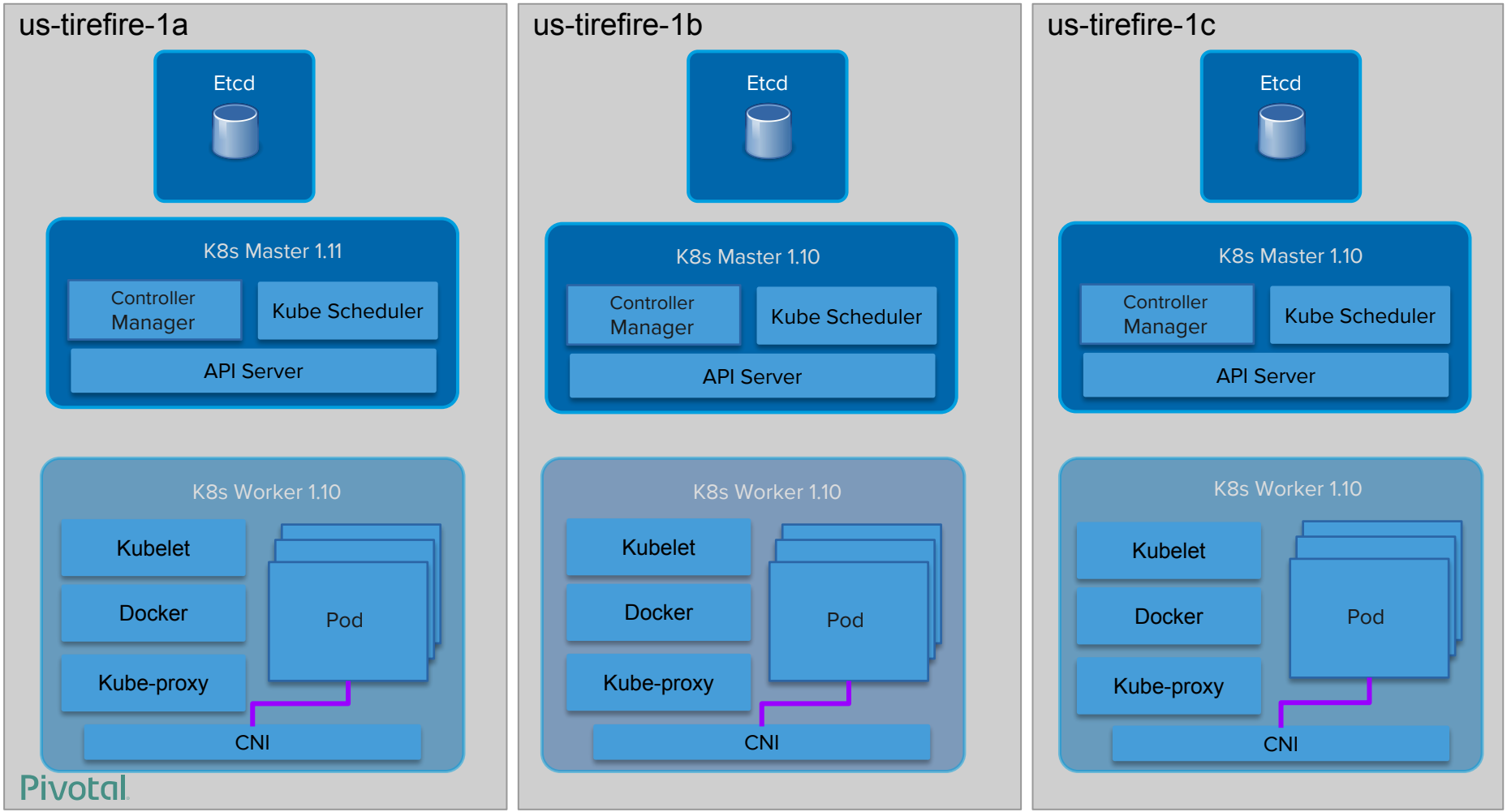
https://github.com/heptio/ark

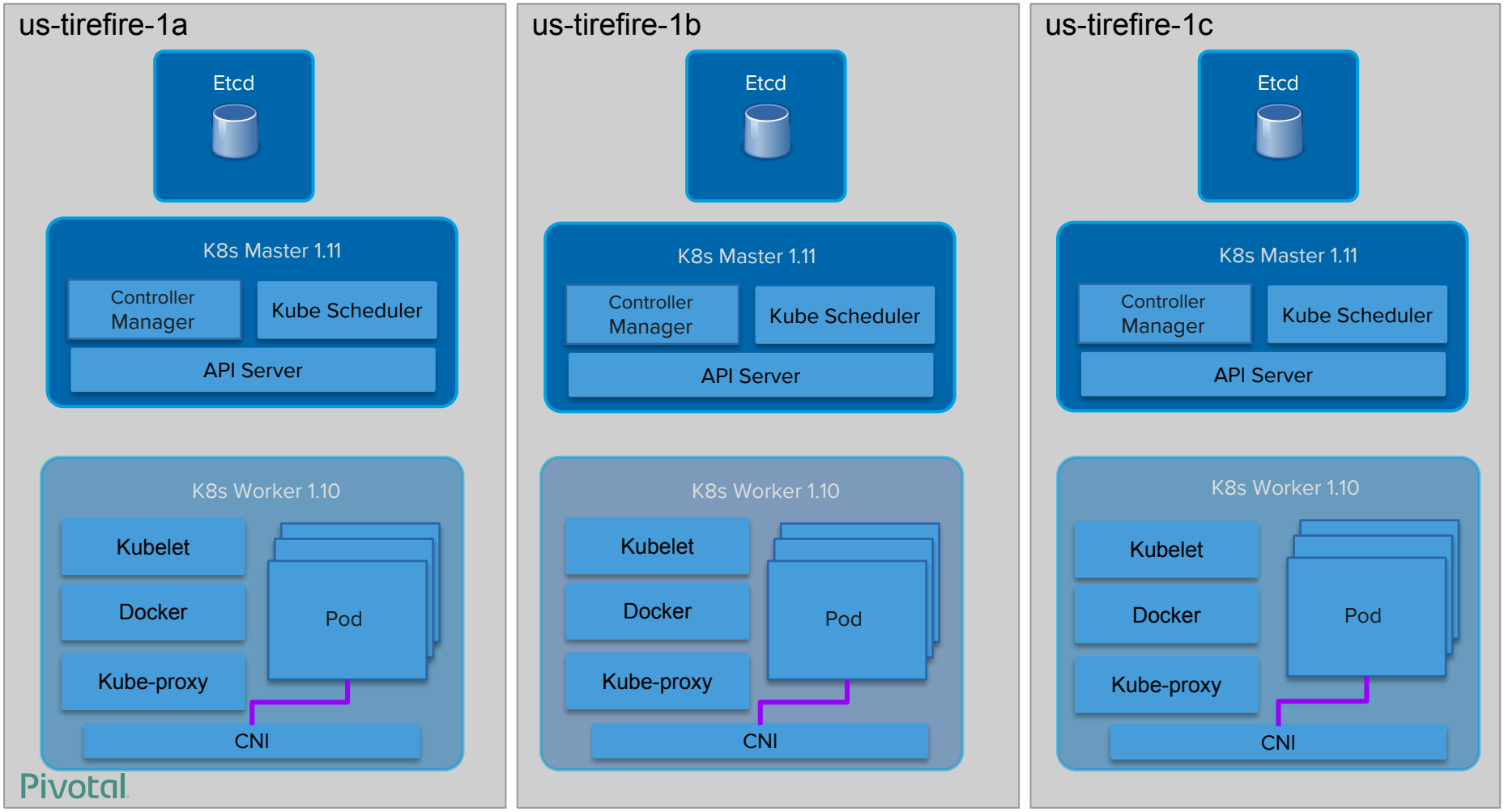

Upgrades

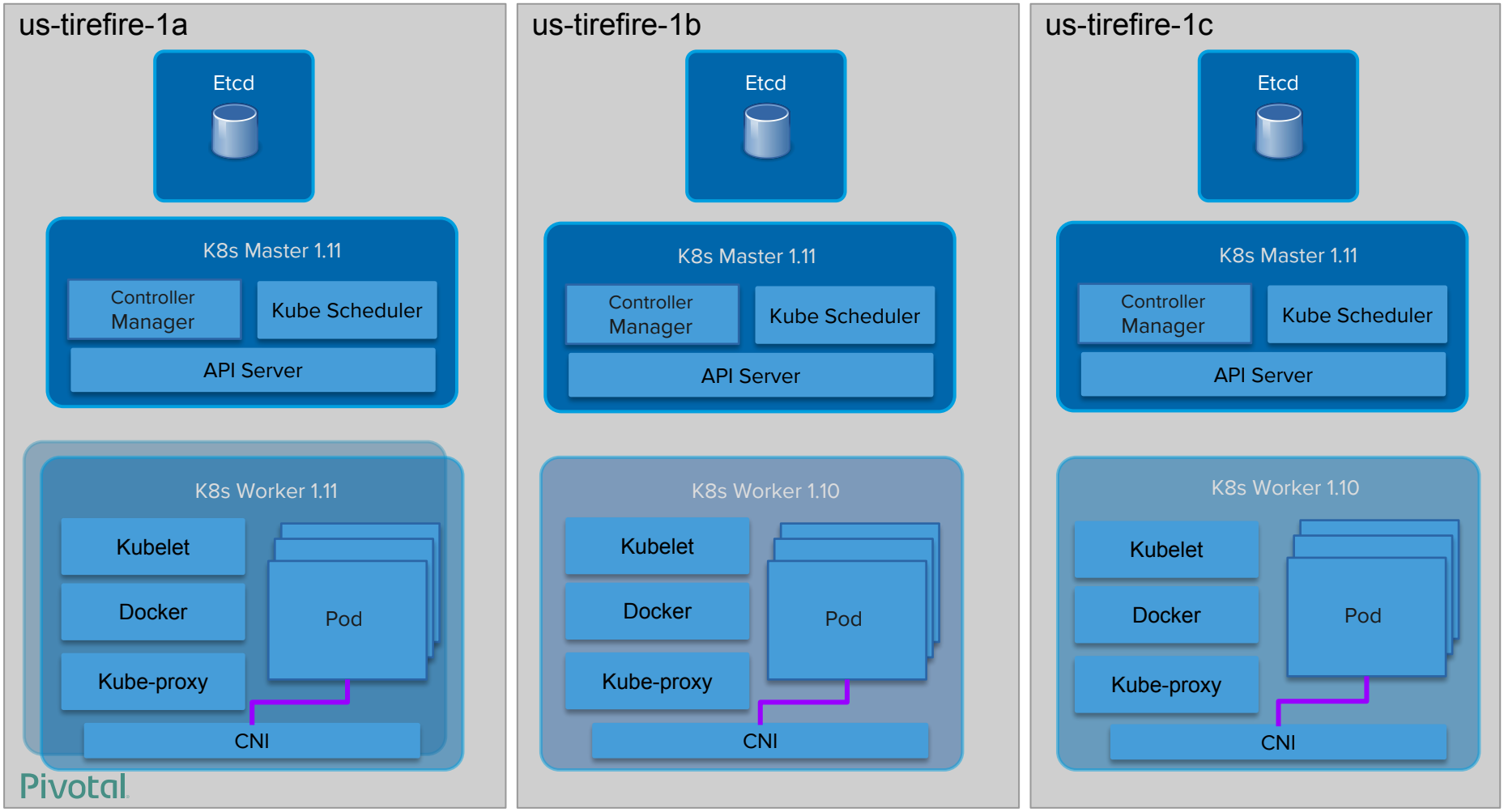

Etcd
K8s Master
Controller Manager
Kube Scheduler
API Server
K8s Worker
Kubelet
Docker
Kube-proxy
Pod
CNI

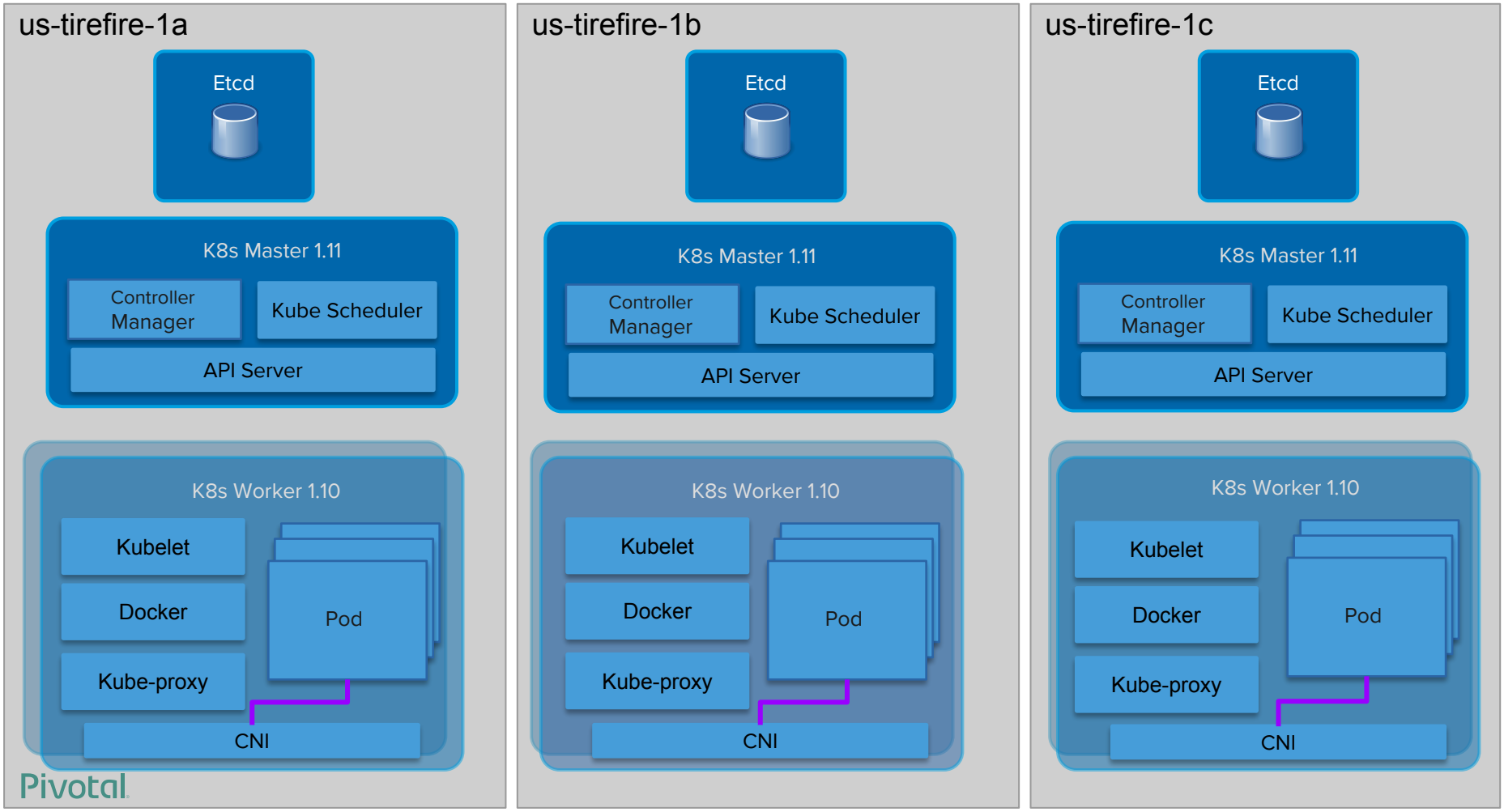

us-tirefire-1a
Etcd
K8s Master 1.10
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1b
Etcd
K8s Master 1.10
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master 1.10
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal

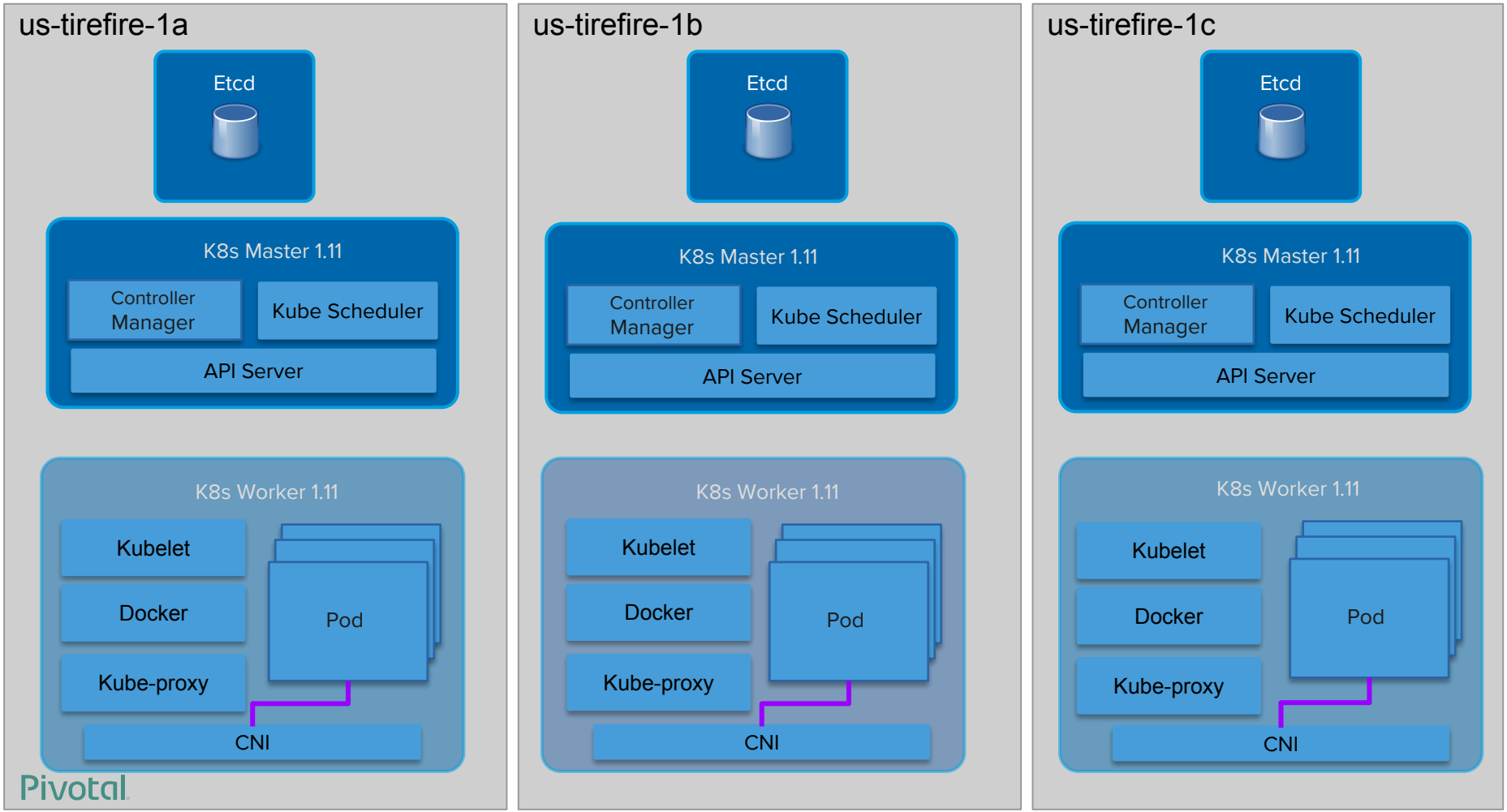

us-tirefire-1a
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1b
Etcd
K8s Master 1.10
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master 1.10
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal


us-tirefire-1a
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1b
Etcd
K8s Master 1.10
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master 1.10
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal


us-tirefire-1a
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1b
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal


us-tirefire-1a
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.11
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal
us-tirefire-1b
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI


us-tirefire-1a
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal
us-tirefire-1b
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.10
Kubelet
Docker
Kube-proxy
Pod
CNI


us-tirefire-1a
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.11
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1b
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.11
Kubelet
Docker
Kube-proxy
Pod
CNI
us-tirefire-1c
Etcd
K8s Master 1.11
Controller Manager
Kube Scheduler
API Server
K8s Worker 1.11
Kubelet
Docker
Kube-proxy
Pod
CNI
Pivotal


Operations

Monitoring / Logging - The Platform

Server Agents

- Install as binaries / containers on the underlying OS
- No chicken and egg problems
- Extra devops toil (config management etc)
- Direct access to system metrics and logs
- Can use existing tools / processes

Daemonsets

- Run in Kubernetes on each node as daemonset
- If Kubernetes is broken, will the monitoring daemonset be broken ?
- Have to be able to dockerize the agent
- Privileged containers / host volumes to access system metrics and logs
- Masters also have to be workers or can’t run daemonset on them.

Monitoring / Logging - Workloads

Kubernetes Metrics API

- Basic point in time pod/node metrics
- $ kubectl top {node,pod}
- Adaptors for Prometheus / Graphite / etc

Kubernetes logging

- Kubernetes configures docker to log all Pod stdout/sterr to a file
- $ kubectl logs <name-of-pod>
- Need daemonset or agent to read k8s logs from filesystem
- EFK - Elastic, Fluent, Kibana

Authentication / Access Control

- Node vs ~~ABAC~~ vs RBAC
- Service Accounts - managed inside kubernetes
- User Accounts - managed outside kubernetes
    - OpenID Connect
        - Ldap / AD
        - Oauth2
        - Etc

- Secure your Kubernetes Dashboard silly!
    - Everything is TLS encrypted .... Right ?

```
$ kubectl auth can-i create deployments --namespace dev
yes
$ kubectl auth can-i create deployments --namespace prod
no
```


FINANCE
benefit
Value!!
Business Company

Replatforming vs Modernization for PKS

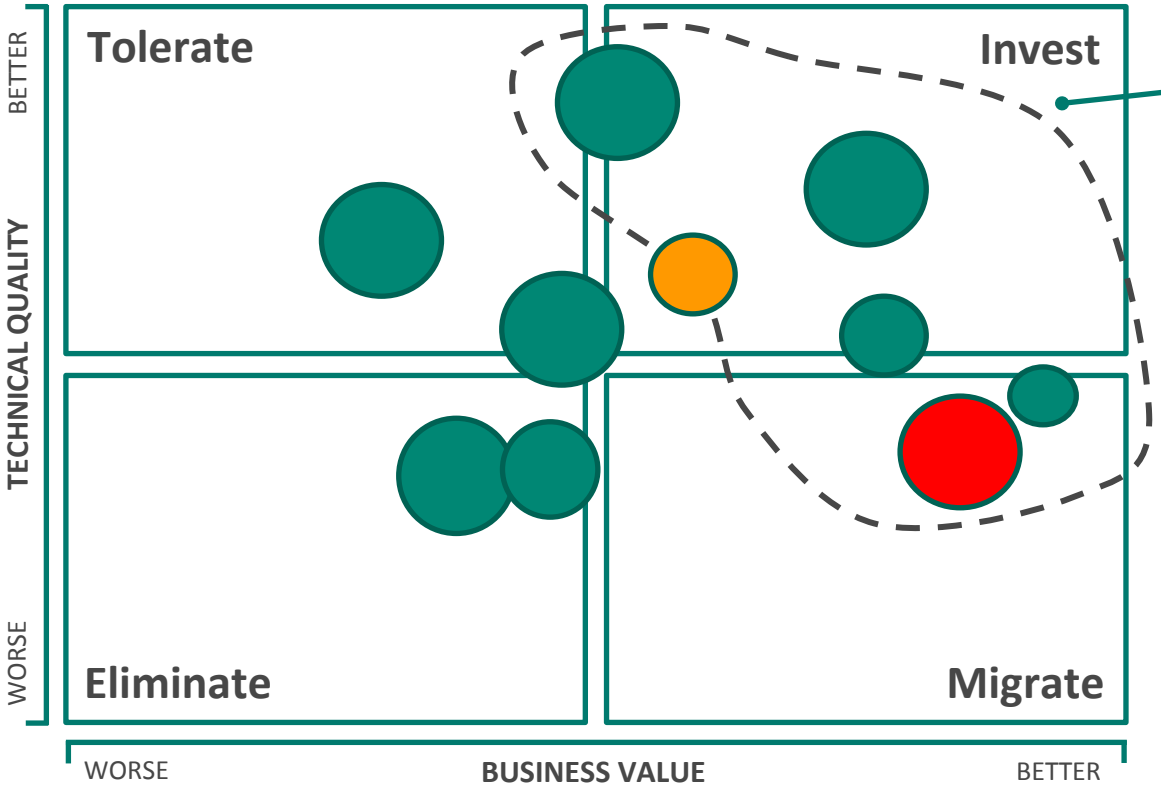
Lift & Shift / Replatforming

- Lift and Shift with "*just enough modernization*"
- You may not have access to the code
- Revisit decisions made in Greenfield time
  - Around CI/CD process
- Get some quick wins through platform capabilities
  - Reduced operating and infrastructure cost
  - Improved speed to deploy & scale
  - Faster patching of kernel level vulnerabilities

Modernization

- Leverage features in modern cloud platforms
  - Blue/Green deploys
  - Auto-healing
  - Auto-scaling
  - Advanced routing/networking automation
- Design and build based on known Cloud Native patterns
- Longer term investment in the application
- Likey you need access to the code
- Plus everything mentioned in "replatforming"

TIME Methodology



Identify top 10s list

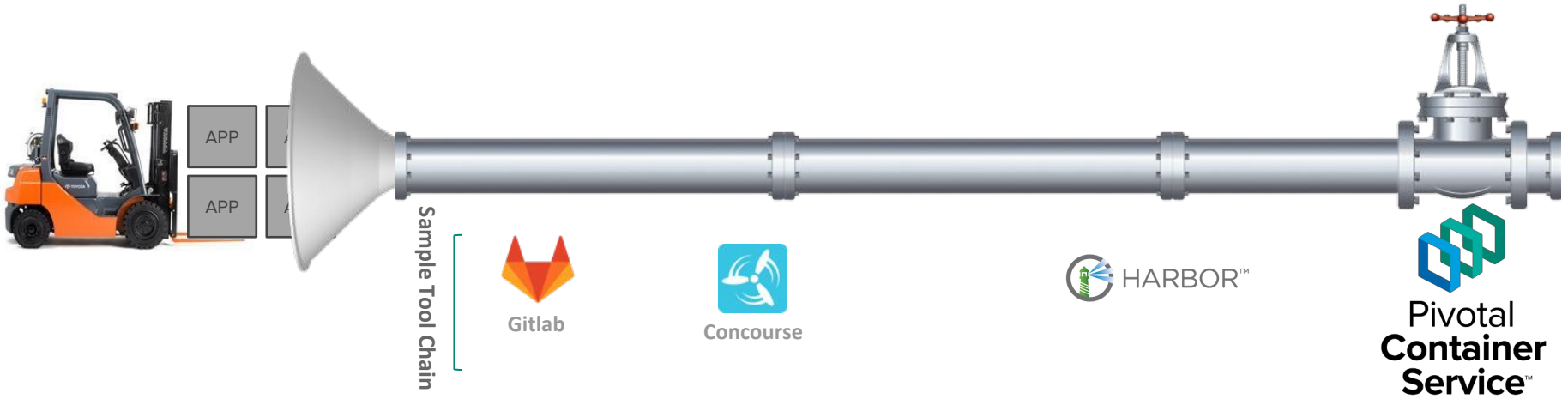
TECHNICAL QUALITY - Technical Debt Level
BUSINESS VALUE - Revenue / Cost Impact

* Gartner's TIME methodology for Application Portfolio Rationalization

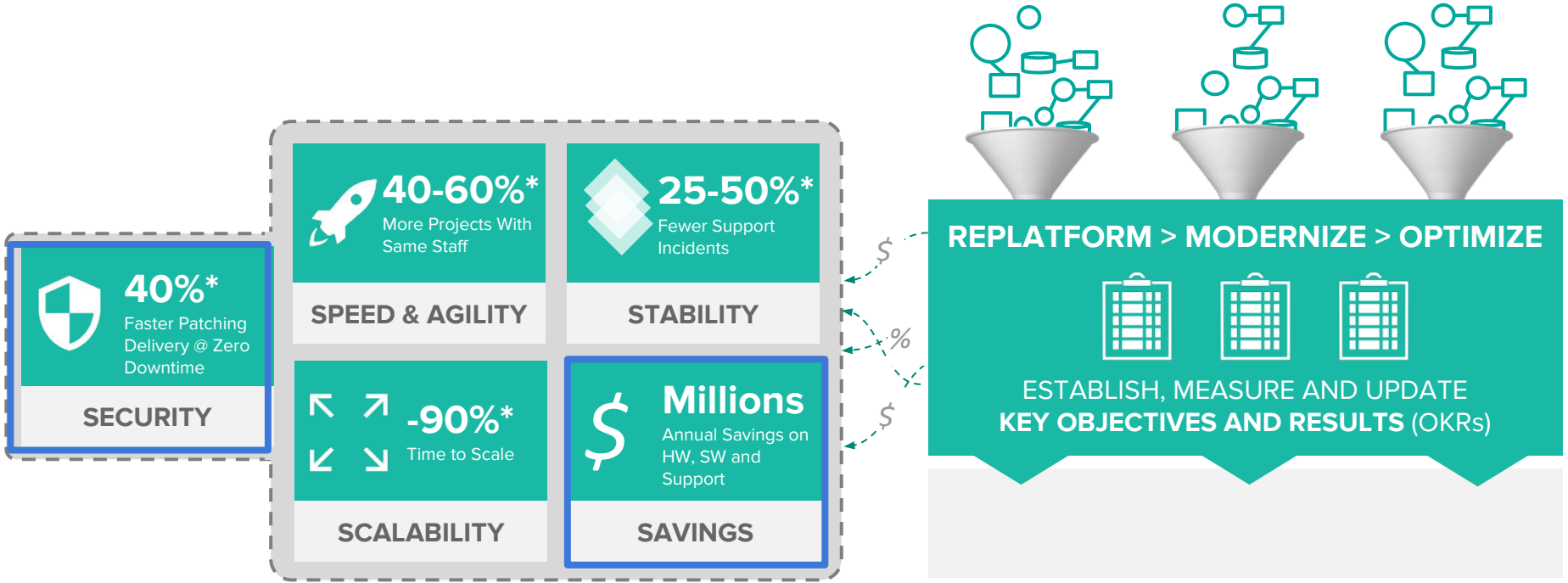
Pivotal

1 **Identify 5-10 apps confirmed as suitable to run on PKS**

2 **Work on a short project to push a few apps *all the way to* prod and measure the ROI metrics**



How We Think about the Business Case



PLATFORM VALUE STREAM AND METRICS

Transforming How The World Builds Software